

IN THE CLAIMS:

1. (Currently amended) A method for carrying out a connection migration and functional continuity of a software application in a multi-computer architecture (cluster), the method comprising:

executing, from a first node, an initial software application on a first computer in the multi-computer architecture, wherein the first node is a called primary node, and wherein the initial software application is executed towards at least one secondary node, wherein the at least one secondary node is on a second computer in the multi-computer architecture; comprising a first computer in said cluster whereon an initial software application is executed, towards at least one secondary node, comprising another computer in said cluster, characterised in that is implements

implementing a virtual network address which is carried by the first computer; and which is transferred to the other computer,

transferring, by the first computer, the virtual network address to the second computer;
providing the virtual network address as a dialogue link to client computers connected to the multi-computer architecture and affected by the initial software application, wherein the initial software application is intended to be replicated on at least one other computer in order to enable a service switch from the initial software application the a replica software application; said virtual network address being provided as a dialogue link between the cluster and clients computers connected to said cluster and affected by the software application

capturing messages issued from a client before the messages are taken into account by a network layer of the multi-computer architecture, wherein the messages are captured at the IP tables level in the context of a protocol TCP/IP;

capturing socket parameters via extended generic system calls;
modifying the socket parameters via the extended generic system calls, wherein the socket parameters that are captured and modified include at least one of local and remote ports, local and remote reference number, number of a next packet to be emitted and expected, emission and reception timer, and window size;

backing up a list of packages pending transmission, packages in transit, and packages received but not yet read by the initial software application;

transmitting the messages to the second computer over a multicast-type channel;
updating, on-the-flow, at least one clone from a plurality of clones installed on the secondary node of the initial software application, wherein the at least one clone is updated periodically, wherein the updating of the at least one clone is triggered by one or more characteristic events, and wherein each clone in the plurality of clones includes a record of the messages received by the primary node;

responsive to detecting a fault or an event affecting the first node, electing the at least one clone from the plurality of clones to be substituted for the initial software application;

switching to the at least one clone, wherein the at least one clone becomes the new primary node, and wherein the record of the messages received by the primary node are re-injected into the new primary node;

responsive to switching to the at least one clone, migrating network connections; and supervising a state of resources necessary to operation of the initial software application.

2.-20. (Canceled)